MODIS DATA STUDY
TEAM PRESENTATION

November 30, 1990

AGENDA

1. Action Items
2. Anchor Point Accuracies
3. Data Flow Diagram Structure
4. Questions/Comments on "Standards and Guidelines for Science Data Processing Software"

ACTION ITEMS:

10/5/90-2 [Doug Hoyt]: Review the interface between MODIS Level-1 processing and the MCST and identify unresolved issues or points requiring clarification. Prepare a list of needed information items and relate each item to a corresponding Level-1 required activity or design requirement. STATUS: Report given at 11/16/90 meeting. Closed.

10/12/90-2 [Watson Gregg]: Prepare a report on MODIS anchor point requirements. Analyze the utility of alternative parameters to describe MODIS observation and solar geometry. STATUS: Report in this week's handout.

10/19/90-1 [John Blaisdell]: Expand introductory material in Earth Model write-up to include broad discussion of MODIS geolocation and need for Earth model. Coordinate with Al Fleig to distribute report. STATUS: Action Item reassigned to Al McKay and Lloyd Carpenter. Open.

10/26/90-1 [John Blaisdell]: Scope a brief error analysis and impact study on the merits of a geoid model as opposed to an ellipsoid. STATUS: Action Item reassigned to Al McKay and Lloyd Carpenter. Open.

11/16/90-1 [Doug Hoyt]: Review MODIS Level-1 data flow diagrams and identify data items potentially provided by the MCST. Provide a list of instrument parameters required to Earth locate MODIS pixels (e.g. detector locations, electronic delays, mirror rotations, etc). STATUS: Open.

11/16/90-2 [Tom Goff and Al McKay]: Review the preliminary version (28 September 1990) of "Standards and Guidelines for Science Data Processing Software" and provide a list of questions and comments. STATUS: Report in this week's handout.

MODIS ANCHOR POINT ACCURACIES

<u>Definition</u>. Anchor points are subsets of the total pixels in a given observation increment (granule) where navigation parameters are directly computed. The parameters determined at these anchor points may be interpolated to provide parameters for the entire observation increment.

<u>Purpose</u>. To reduce the size of archived data and the number of computations required for data processing. The anchor point method was used for CZCS and AVHRR processing.

<u>Method</u>. The observation increment used in the analysis of anchor point accuracies is the "scan plane", consisting of the across-track by along-track dimensions of a single scan. These are 1354 by 8 for MODIS-N and 1007 by 30 for MODIS-T.

First, each pixel in the scan plane was explicitly georeferenced. Georeferencing included latitude, longitude, solar zenith and azimuth angles, and spacecraft zenith and azimuth angles. The method used here was a short-cut to the method likely to be used for actual geolocation. The method here used the following input parameters, and the values shown.

Input Parameter	MODIS-N	MODIS-T
Altitude	705 km	705 km
Inclination	98.25°	98.25°
Period	98.9 min.	98.9 min.
Julian day	80	80
Equator crossing time	1:30PM	1:30PM
Equator crossing longitude	0°	0°
Time traveled since most recent equator crossing	0 min.	0 min.
Pitch	0°	0°
Roll	0°	0°
Yaw	0°	0°
Tilt	0°	0-50°
IFOV	1 km	1.1 km
Scan width	55°	45°

Three anchor point scenarios -- where the number of anchor points = 2%, 1%, and 0.5% of the total pixels in a scan -- were examined. These test scenarios were applied for both MODIS-N and MODIS-T. The analyses shown here involved a uniform anchor point distribution strategy (equal pixel spacing between anchor points), and a non-uniform strategy (unequal spacing). The non-uniform strategy was designed to produce equal areal coverage across-track, taking into account the increase in area of the pixels at the scan edges relative to nadir. The across-track, along-track pixel numbers and increments for each sensor for each scenario are shown below.

For each scenario we examined errors, computations, and storage under a linear interpolation method and a cubic spline interpolation method. The cubic spline interpolation became a linear interpolation along-track for MODIS-N because there were only two anchor points along-track for MODIS-N.

We assumed that the number of computations to explicitly locate a pixel was 450, based on the estimates by the MODIS Data Study Team (January 26, 1990 Report). This estimate is assumed to include all 6 variables computed here. The method of computation of these floating point operations is given in the Appendix. We did not take into account problems in interpolation near the poles and the date line in these estimates. For the storage requirements, we assumed each variable was a 32-bit length word. Also, we included the computer run-time for these calculations. All simulations were performed on a 386-25 MHz computer containing a math coprocessor. The linear interpolation and explicit computations had the same number fo input/output operations, but the cubic spline had only 1/3 this amount due to memory constraints.

All analyses were performed for a single scan at the equator.

<u>Results</u>. Results are depicted in the following tables. Included are analyses for MODIS-N and MODIS-T under two tilts, 0° and 50°. Calculations for explicit Earth location are included in the tables for comparison.

The advantages of anchor points are clearly demonstrated by the reduction in number of computations, storage, and computer run-time over explicit Earth location. Even for the worst case (2% anchor points), linear interpolation of anchor points resulted in a factor of 27 reduction in number of computations, a factor of 52 reduction in storage, and a slight reduction in computer run-time for MODIS-N and T. Cubic spline interpolation produced no benefit in storage over linear interpolation, and slightly increased the number of computations, but produced a major reduction in computer run-time (factor of 3.5 for MODIS-N and factor of 2.8 for MODIS-T). However, since the number of I/O operations for the cubic spline was reduced by 2/3, these numbers are not representative of run-time savings.

The disadvantage of anchor points is a loss of accuracy, regardless of the interpolation method and distribution. However, if this accuracy can be brought down to acceptable limits, the storage and computational efficiency advantages prove desirable. It is this accuracy which is the issue of the present report.

First, it should be noted that errors in solar zenith angle never exceeded 0.2° under any circumstances, nor did solar azimuth angle errors exceed 1.5°. The maximum errors occurred for the 50° tilted MODIS-T case, and occurred near the extreme right-hand (eastern) edge of the scan. Since this position corresponds to the highest solar zenith angle for the scan, we decided to test the errors for a scan crossing the terminator. In this circumstance, there was no increase in errors of the solar angles, suggesting that the accuracy of anchor points in solar angles is not an important factor in the decision whether or not to use anchor points.

Similarly, the use of anchor points produced errors in spacecraft zenith angle that did not exceed 2.3°. This maximum error occurred for MODIS-N using 5% anchor points and a uniform distribution. It is noteworthy that in all cases the maximum errors occurred within 1 pixel of nadir. Since at nadir the spacecraft zenith angle is small, and variations about this angle are relatively unimportant for scattering calculations, these small errors are even more unimportant than their small sizes suggest.

As with spacecraft zenith angle, the maximum errors in spacecraft azimuth angle occurred within 1 pixel of nadir. Despite the apparently large values of the azimuth errors, they are even less important in scattering calculations near nadir than the zenith angle. Just off nadir, where an error in azimuth angle can have considerable importance in scattering calculations, the errors were reduced to < 1°. Thus the large errors are misleading and the use of anchor points is justified with respect to spacecraft azimuth angle.

It is in the latitudes and longitudes that the case for anchor points must win or lose. It is here that the errors are the largest, and where the different strategies have the most impact. Errors in longitude are always greater than errors in latitude, due to the orientation of the scan plane with respect to the Earth -- at the equator the sensor is scanning mostly across longitudes with little change in latitude.

As a generality it is apparent that a non-uniform anchor point distribution reduces errors in longitude and latitude relative to a uniform distribution, and that cubic spline interpolation reduces errors further still. Since a non-uniform distribution produces **no** cost in computations, storage, or speed, it is clearly the better choice of distribution strategy. Secondly, since cubic spline interpolation produces only minor increase in floating point operations, no gain in storage, and probably a minor increase in speed, it is the better interpolation method. **Assembling these facts, a cubic spline interpolation of non-uniformly distributed anchor points is the best anchor point strategy** of the methods studied here.

If we confine ourselves to this approach, we see that there is little difference in the errors for a 2% anchor point density compared to a 1% density, for MODIS-N and an untilted MODIS-T.

There is, however, a factor of 2 increase in longitude error for MODIS-N for a 0.5% density. Considering the reduced storage requirements for a 1% density compared to a 2% density, the 1% density is preferred for untilted sensors.

However, for a "maximally" tilted (50°) MODIS-T the errors for a 1% density are > 3 times that for a 2% strategy. The errors for even a 2% density are unacceptably large, however, despite the fact that they represent ≈ 12% of the pixel size at this location. These large errors are a consequence of the fact that pixels are distorted along-track at this tilt in addition to across-track. The anchor point distribution strategies we used here focused on resolving the across-track errors. This is an acceptable approach for a non-tilting or minimally tilting (e.g. ±20°) sensor. But it cannot resolve the along-track errors for a sensor capable of the tilts of MODIS-T. The solution is to reduce the number of across-track anchor points and increase the number of along-track anchor points.

<u>Conclusion</u>. The use if anchor points reduces the computational and storage burden of the data system, and increases the computation speed. If anchor points can produce acceptable accuracies in Earth location, then their use is advantageous. Anchor point densities of 2% and 1% can potentially produce these accuracies for non-tilting sensors (i.e. MODIS-N), and thus a 1% density is preferred because of reduced storage. However, to keep errors within acceptable limits for high tilts (≈50°), more anchor points are required in the along-track direction than studies here. Whether this requires a higher density than 2% is a question for further study.

Table 1. Along-track, across-track anchor point pixel numbers and increments (anchor point distance in pixels) for different anchor point location strategies and percents. In each case, along-track is shown first, as in

along-track
across-track

MODIS-N		Number	Increment
Uniform Distribution	2%	2 105	7 13
	1%	2 55	7 25
	0.5%	2 28	7 50
		Number	**Increment**
Non-Uniform Distribution	2%	2 106	7 4 (min) 22 (max)
	1%	2 56	7 8 (min) 42 (max)
	0.5%	2 28	7 17 (min) 80 (max)

MODIS-T		Number	Increment
Uniform Distribution	2%	4 145	8 7
	1%	3 101	15 10
	0.5%	3 51	15 20
		Number	**Increment**
Non-Uniform Distribution	2%	4 146	8 3 (min) 8 (max)
	1%	3 102	15 5 (min) 13 (max)
	0.5%	3 51	15 10 (min) 25 (max)

SENSOR	MODIS-N	TILT	0															
	LINEAR MAXIMUM ERROR									CUBIC MAXIMUM ERROR								
	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp
UNIFORM 2%	104	737	0.007	0.02	0.3	326	0.18	5.0	65	9	30	0.0003	0.0006	0.3	335	0.21	5.0	20
UNIFORM 1%	348	2468	0.02	0.06	0.4	313	0.14	2.6	65	15	105	0.0009	0.005	0.3	321	0.17	2.6	20
UNIFORM 0.5%	1190	8392	0.08	0.2	2.3	-181	0.11	1.3	65	105	753	0.007	0.03	1.5	182	0.14	1.3	20
NON-UNIFORM 2%	20	135	0.001	0.003	0.8	171	0.18	5.0	65	4	22	0.0002	0.0001	0.6	171	0.21	5.0	20
NON-UNIFORM 1%	64	448	0.004	0.01	1.9	172	0.14	2.6	65	4	24	0.0002	0.0002	1.3	173	0.17	2.6	20
NON-UNIFORM 0.5%	220	1558	0.01	0.03	0.4	-343	0.11	1.3	65	7	51	0.0005	0.002	-1.1	344	0.14	1.3	20
EXPLICIT							4.9	260	70							4.9	260	70

Ψ = latitude (m)
Φ = longitude (m)
θ_o = solar zenith angle (°)
ϕ_o = solar azimuth angle (°)
θ = spacecraft zenith angle (°)
ϕ = spacecraft azimuth angle (°)
F = number of computations (mega floating point operations/scan)
S = storage (kbytes)
Sp = speed (sec) on 386-25 computer to compute all pixels

SENSOR	MODIS-T	TILT	0

	LINEAR MAXIMUM ERROR									CUBIC MAXIMUM ERROR								
	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp
UNIFORM 2%	15	97	0.0009	0.001	0.5	300	0.50	13.9	170	0.6	0.6	0	0	0.3	344	0.55	13.9	70
UNIFORM 1%	31	204	0.002	0.002	0.5	-304	0.38	7.3	170	2	1	0	0	0.3	-320	0.42	7.3	70
UNIFORM 0.5%	89	601	0.006	0.008	0.5	285	0.31	3.7	170	3	6	0	0.002	-0.3	302	0.34	3.7	70
NON-UNIFORM 2%	9	56	0.0006	0.0004	0.6	-216	0.50	13.9	170	-0.6	0.6	0	0	0.4	-222	0.55	13.9	70
NON-UNIFORM 1%	20	119	0.001	0.001	0.7	-195	0.38	7.3	170	0.8	2	0	0	0.5	-197	0.42	7.3	70
NON-UNIFORM 0.5%	102	697	0.006	0.003	1.3	-191	0.31	3.7	170	2	-2	0	0.0001	0.9	-192	0.34	3.7	70
EXPLICIT							13.6	725	175							13.6	725	195

Ψ = latitude (m)
Φ = longitude (m)
θ_o = solar zenith angle (°)
ϕ_o = solar azimuth angle (°)
θ = spacecraft zenith angle (°)
ϕ = spacecraft azimuth angle (°)
F = number of computations (mega floating point operations/scan)
S = storage (kbytes)
Sp = speed (sec) on 386-25 computer to compute all pixels

SENSOR	MODIS-T	TILT	50															
	LINEAR MAXIMUM ERROR									CUBIC MAXIMUM ERROR								
	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp	Ψ	Φ	θ_o	ϕ_o	θ	ϕ	F	S	Sp
UNIFORM 2%	8657	11827	0.1	-0.6	0.1	0.03	0.50	13.9	170	1414	2025	0.02	-0.08	0.02	0.004	0.55	13.9	70
UNIFORM 1%	14097	19076	0.2	-1.0	0.2	0.04	0.38	7.3	170	5436	7479	0.06	-0.3	0.08	0.02	0.42	7.3	70
UNIFORM 0.5%	19166	27422	0.2	-1.5	0.3	0.06	0.31	3.7	170	5907	8332	0.07	-0.4	0.08	0.02	0.34	3.7	70
NON-UNIFORM 2%	7974	10642	0.09	-0.5	0.1	0.002	0.50	13.9	170	1414	2025	0.02	-0.08	0.02	0.004	0.55	13.9	70
NON-UNIFORM 1%	13289	17609	0.2	-0.8	0.2	0.04	0.38	7.3	170	5436	7479	0.06	-0.3	0.08	0.02	0.42	7.3	70
NON-UNIFORM 0.5%	14097	19076	0.2	-1.0	0.2	0.04	0.31	3.7	170	5436	7479	0.06	-0.3	0.08	0.02	0.34	3.7	70
EXPLICIT							13.6	725	195							13.6	725	195

Ψ = latitude (m)
Φ = longitude (m)
θ_o = solar zenith angle (°)
ϕ_o = solar azimuth angle (°)
θ = spacecraft zenith angle (°)
ϕ = spacecraft azimuth angle (°)
F = number of computations (mega floating point operations/scan)
S = storage (kbytes)
Sp = speed (sec) on 386-25 computer to compute all pixels

Appendix. Method of computation of number of floating point operations.

	MODIS-N	MODIS-T
P = number of pixels across-track	1354	1007
D = number of detectors along-track	8	30
E = number of floating point operations to locate one pixel	450	450
N = number of anchor points	2%, 1% or 0.5%	
A = number of anchor points along track	2	3 or 4
R = total number of floating point operations		

Bi-Linear Interpolation -- requires 8 operations

$$R = E*N + P*D*8$$

Cubic Spline Interpolation

A. Set-up -- requires 24 operations at each anchor point

$$R_1 = E*N + 11 + 24*N$$

B. Interpolation -- requires 10 operations
R_2 = along-track interpolation
R_3 = across-track interpolation

$$R_2 = (10P)A$$
$$R_3 = P*D*10$$

CASE Symbol Descriptions

This is a description of the various symbols used on the Sample Transaction Diagram, an enhanced version of a Data Flow Diagram.

DATA PROCESS - This function acts upon, extracts, or appends item in a flow of data. This is the common data flow diagram processor.

CONTROL TRANSFORM - This is a control function only. It accepts and/or generates control specifiers that may change the mode (for example) of operation of the Data Process, but not necessarily the function of the Data Process. This item transforms the Process from one method into another method.

DATA FLOW - The functional (not necessarily physical) passing of data from one process to another.

CONTROL FLOW - The passing of control specifiers (state variable, case indicators) to/from Data Processes.

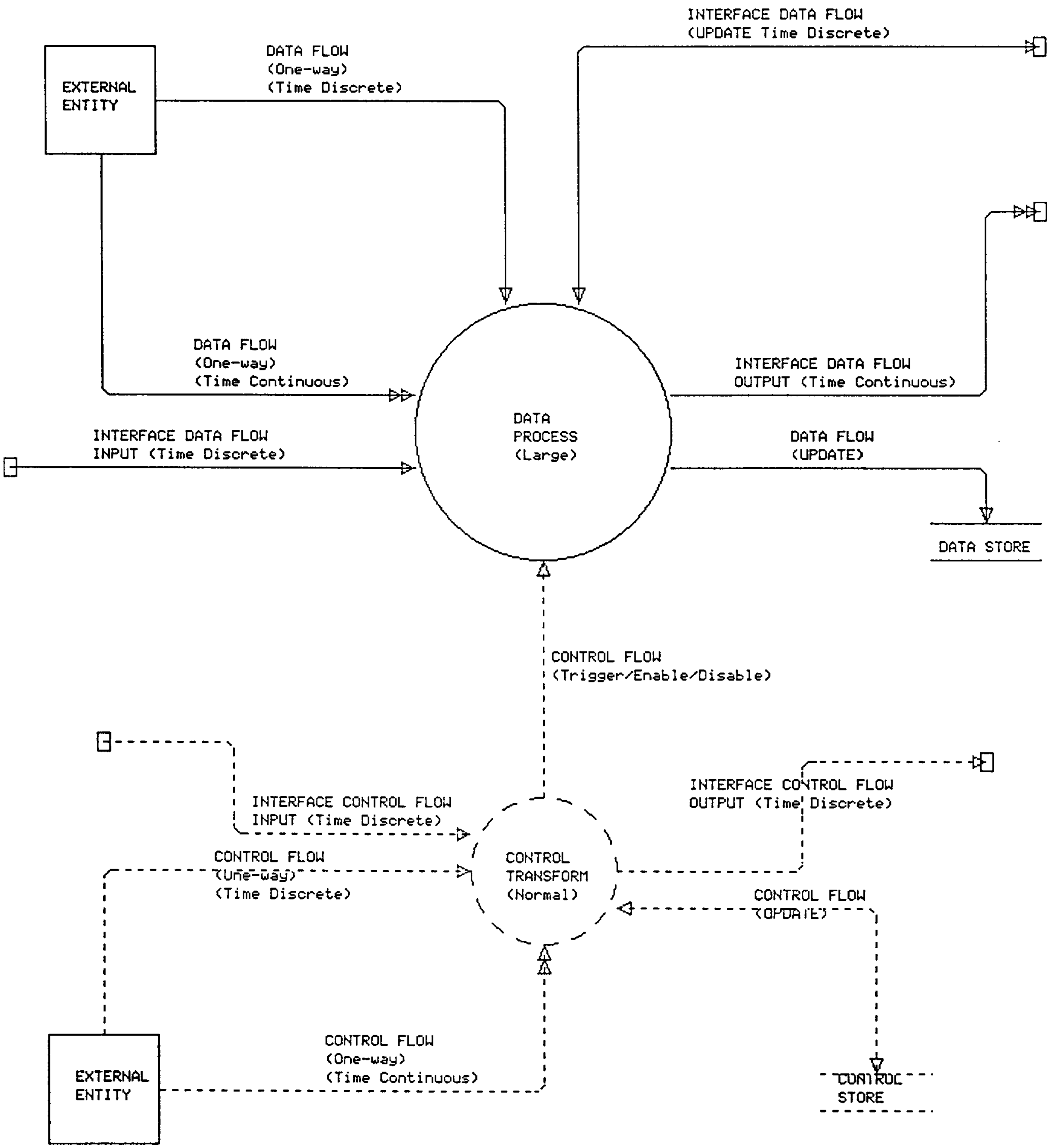
FLOWS can be uni- or bi-directional, and discrete or continuous. Continuous flows do not exist in the MODIS processing.

DATA STORE - A localized storage of data items to be used by more than one processor (bubble).

CONTROL STORE - a localized store to be used by Control Transforms only! This allows non-telemetered or similiar commands that build upon provious states to be determined.

TIME DISCRETE - The Flow item is available when requested. There is a one-to-one correspondence between available data and the processing of that data. Example: handshaked data request/response.

TIME CONTINUOUS - The Flow item value is always available but may only be sampled as necessary. A lack of handshaking is implied. Example: a voltage measurement by an A/D converter.



Sample Transaction Diagram

Questions/Comments on "Standards and Guidelines for Science Data Processing Software"

1. As written, the document provides project management guidelines for software development but does not adequately address specific features of desired software. The document would perhaps best be labeled as a software management guide and not a software standards specification.

2. Requirements in the document seem to exclude the entire science community. From page 1 of the Executive Summary, "These policies and guidelines pertain to all software except for the set of routines responsible for performing the mathematical manipulations of the input (the 'science algorithms')".

3. To facilitate the development of machine-independent software, EOSDIS should provide ANSI-compatible sofware development environments for a number of machines. Individual science investigators should not be left "on their own" to select compatible environments suitable for EOSDIS purposes. Rather than simply requiring investigators to develop code that will run on EOSDIS facilities, the project should provide a development environment or environments that will automatically ensure that developed code is suitable for EOS purposes.

4. Given the size of the EOSDIS undertaking, software documentation techniques that provide broad overviews and easy insight to large and complex systems is required. The document alludes to several such techniques but does not recommend a specific technique for universal use. Without a standardized technique, a user may have to learn several techniques and provide mental translations from one description language to another. The designation of standard software development tools may facilitate the development of consistent and readily understood software documentation by a large group of heterogeneous developers.

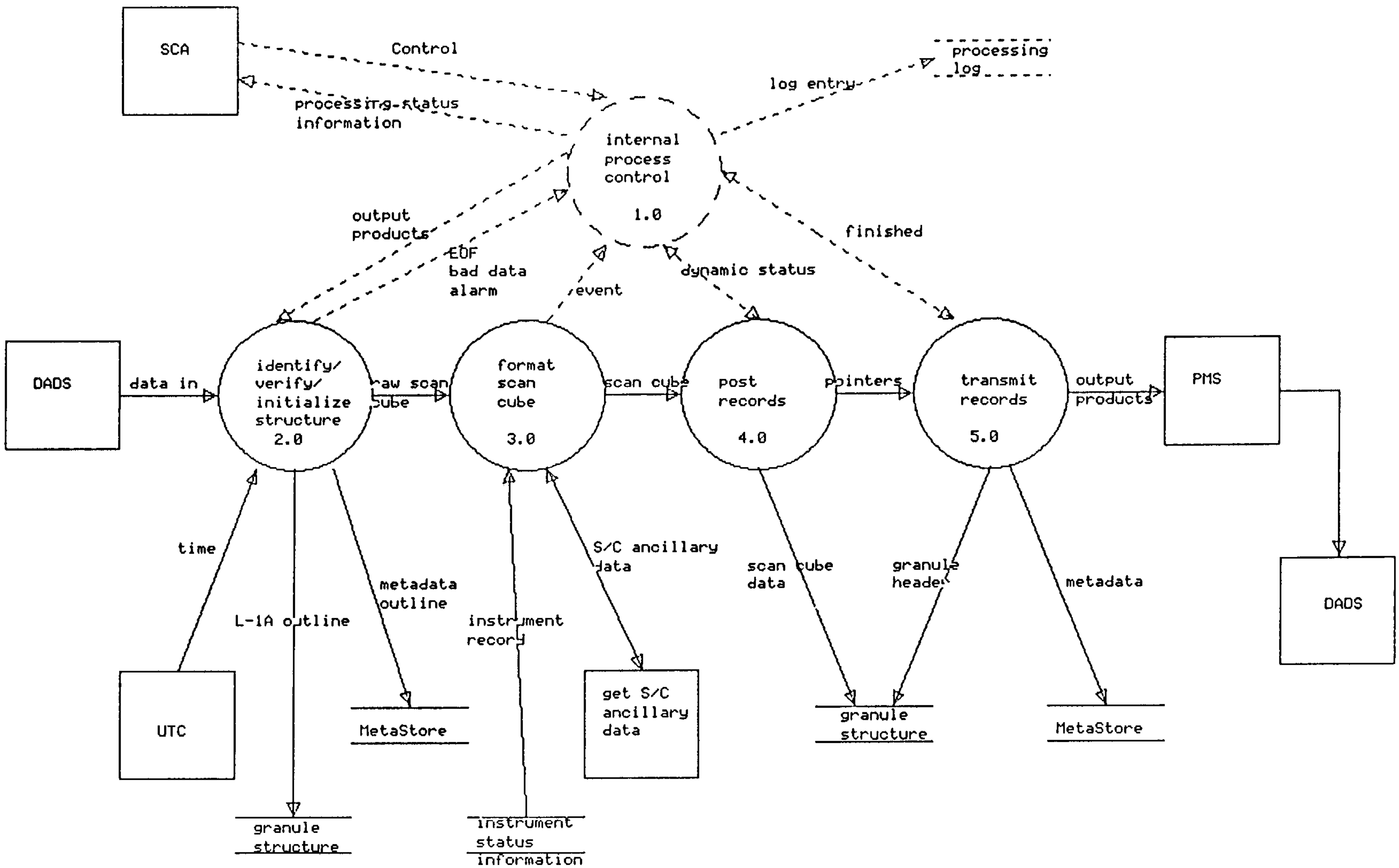
5. The FORTRAN guidelines provided in Appendix C seem very dated and do not account for developments of the last fifteen years. In particular, code documentation structures that facilitate automated search and retrieval are very valuable. Other automated code management techniques are also useful.

MODIS DATA STUDY TEAM APPENDIX:

LEVEL-1A DATA FLOW DIAGRAM AND DATA DICTIONARY

Including Assumptions List and
Issue for Discussion

NOVEMBER 30, 1990



MODIS-1A Design
Level A

Data Dictionary for the Software Engineering of the MODIS-1A Product Generation Program

This data dictionary document (DDD) contains all items as shown on the various data flow diagrams that accompany this document. All items in the context diagram, data flow diagram, event list, entity relationship diagram, state transition diagram, or other diagrams will be included here. See the appendix at the end of this document for definitions and examples of the content titles.

Name Type From To : Description

Control Message SCA InternalProcessControl : Messages that inform the process which mode to operate in, to start, to stop, to suspend, to resume, to return status (dynamically or statically), and to request and verify the staging and/or destaging of data to/from the DADS.

DADS : Data Archive and Distribution System

DataIn DataFlow DADS IdentifyVerifyInitializeStructure : Level-0 data or quick-look data.

DynamicStatus ControlRequest/Message InternalProcessControl PostRecords : Two way dialog asking for current dynamic status and returning this status information.

EofBadDataAlarm ControlMessage IdentifyVerifyInitializeStructure InternalProcessControl : Signals an end of data input, signals bad or inappropriate data, requests an alarm generation.

Event ControlMessage CreateProduct InternalProcessControl : Anomaly in instrument status between IC log and telemetered data.

Finished ControlMessage InternalProcessControl TransmitRecords : Request graceful termination (post data) : granule is filled up, terminate.

FormatScanCube DataProcess : Checks instrument status indicators, byte aligns science data, appends S/C platform ephemeris and attitide, and updates packet accounting.

GetS/C_AncillaryData ExternalProcess : An external process (subroutine or separate program) that returns the S/C platform ephemeris and attitude data records in the neighborhood of the given S/C time.

GranuleHeader DataStoreRecord TransmitRecords GranuleStructure : Information about the data granule that is necessary for subsequent data processing or for understanding the data in the granule.
An identifier for this data set.

GranuleStructure DataStore : Description of the level-1A granule in processor memory. Initialized with a 'bad data' indicator and filled with 'good data' as it is processed.

IdentifyVerifyInitializeStructure DataProcess : Verify packet identity, examine level-0 data quality fields, set routing control flags, set up data store areas.

InstrumentRecord StoreRecord InstrumentStatusInformation FormatScanCube : A snapshot of the instrument state at a specified time; integrated from previous instrument commanding.

InstrumentStatusInformation DataStore : Instrument status at a given time as determined by the Instrument Control Center. To be compared with the telemetered status. This is the integrated result of all previous commands.

InternalProcessControl DataProcess : The control functions of the MODIS processor.

L-1A_Outline DataStoreRecord IdentifyVerifyInitializeStructure GranuleStructure : The definitive parameters of the memory requirements for the elements of the data product granule structure (template); the result of a request for operating system services. The structure data area is initialized to a 'bad data' indication.

LogEntry Message InternalProcessControl ProcessingLog : Blow by blow of processing status, time sequential.

Metadata DataStoreRecord TransmitRecord Metastore : Information derived from data sets that provides an understanding of the content or utility of that data set. Updated constantly.

MetadataOutline DataStoreRecord IdentifyVerifyInitializeStructure Metadata : The definitive parameters of the memory requirements for the elements of the metadata structure; the result of a request for operating system services. The structure metadata is initialized to a 'bad data' indication.

MetaStore DataStore : Processor memory allocated for the placement of the metadata items. Initialized to 'invalid'.

OutputProducts DataFlow TransmitRecords PMS : Generated product or pointers to products, where products may consist of: level-1A instrument data with header and data quality information, level-1A metadata, and/or quick-look product.

PMS : Product Management Service

Pointers DataFlow PostRecords TransmitRecords : Location and size of the data structures; assess completeness of structure.

PostRecords DataProcess : Place scan cube into the data granule, accounting for time-ordering of non-time-ordered quick-look data.

ProcessingLog ControlStore : Log of processing status records, time sequential.

ProcessingStatusInformation Message InternalProcessControl SCA : Information reqarding the fault conditions and processing performance of the data processing system. Status or completion information from the MODIS process to the SCA.

RawScanCube DataFlow IdentifyVerifyInitializeStructure CreateProduct : Packet data that has been placed into a cube type record containing across track pixels on the x-axis, along track pixels on the y-axis, and wavelength on the z-axis.

S/C_AncillaryData GetS/C_AncillaryData FormatScanCube DataFlow : Platform ephemeris and attitude data within the time neighborhood of the requested time. I.e. the ten time-tagged platform ancillary records surrounding the scan cube; asynchronous with the scan cube, not interpolated.

SCA : Scheduling, Control, and Accounting

ScanCube DataFlow FormatScanCube PortRecords : Cube of scan oriented data formatted to the data product specification.

ScanCubeData DataStoreRecord PostRecords GranuleStructure : Cube of MODIS data without header, instrument status, and S/C ancillary appended.

StartStop ControlMessage InternalProcessControl IdentifyVerifyInitializeStructure : Starts the process with initialization parameters, panic stop executing.

Time DataFlow UTC IdentifyVerifyInitializeStructure : True universal coordinated time.

TransmitRecords DataProcess : Access product completeness, find location within structure of product components, transmit properly sequenced components to the PMS. (This can be either the actual data or pointers to the data - TBD)

UTC : Universal Time Coordinated

Appendix

NOTES: The form of this DDD is generated to allow computer sorting of the fields for the Name, Type, From, and To specifiers. Each item is to be processed as a record (no hard c/r's) with white space used as delimiters. Field items must be one character string delimited by white space. These fields are case sensitive, beware!

Name:= formal name of the item, e.g., CDOS-input-data

Type:= item type and heirarchy, examples: data-store-element, alias:nnn, schedule-parameterss-composite

From:= flow originating from this segment, example: PGS

To:= flow being passed to this segment.

Description:= full description of the item including: response times, organization, access restrictions and purpose. Is it sequential, repeated, one-of-many, or a combination. Include forward and backward pointers to other hierarchies in a composite item, and cross referenced.

Assumptions and concerns are documented in separate documents that are to be referenced (pointers) by this document where appropriate.

Justification for the existence of structure items will be referenced to the appropriate requirements document or otherwise justified by reference if at all posible.

This file is located on: \EasyCASE\ModIS\DD.wp

ASSUMPTIONS LIST FOR LEVEL 0 OF MODIS LEVEL-1A PROCESSING SYSTEM

1. Data will be broken out and stored as granules with a granule header. These granules are larger than the scan cube but no larger than an orbit.

Justification: Many data processing activities are facilitated by the creation of data granules of reasonable size -- memory and storage can allocated, and processing software is easier to write and handle. Metadata, a required output product, will be in granule format in order to describe a coherent part of the data. So granules must be created at Level-1A anyway. Reasonably-sized granules also facilitate the recovery of data quality information, particularly the important data completeness and existence parameters. Such granules have been used for many satellite sensors, with apparent success. Finally, and perhaps more importantly, reasonably-sized granules are convenient, both in the data system design but also to users, who are adjusted to operating with coherent sets of data.

ISSUE: WHETHER TO BYTE-ALIGN (UNPACK) MODIS DATA AT LEVEL-1A.

Pro: Byte-alignment is one of the most computationally expensive parts of the data processing system. It is also difficult for users to develop and use software for unpacking. By unpacking data at Level-1A we will greatly increase the usefulness of the data to users and avoid the costly repeat unpacking that users of Level-1A data must perform each time they order Level-1A data. Since Level-0 data is stored at CDOS there is no need for Level-1A to be used as backup. Even if Level-0 is not stored it is no difficulty to re-create it from the Level-1A (as is the requirement of Level-1A). The probability of software errors in the unpacking algorithm are extremely unlikely since it is easy to test for errors prior to launch -- simply process back-and-forth from Level-0 to Level-1A multiple times under multiple test cases. It is unlikely that any other events will occur that will demand re-processing of Level-1A data. Finally, even if Level-0 data were not stored anywhere and the chance existed for a software error that would make Level-1A invalid, it is better that the importance of error-free code be known in advance and written by those aware of it, and thoroughly tested, than to not unpack. Although unpacking data will increase the storage requirements, this can be rectified by data compression techniques in either hardware or software with no loss of performance or additional processing complexity (de-compressing is performed off-line).

Con: Although CDOS has said they will store Level-0 data, there is a chance they will not have sufficient funds. Since we are not proposing to store Level-0 either, that will leave Level-1A as the ultimate source of MODIS data. There is a possibility that a software error could occur in the unpacking algorithm (or at another critical point in the Level-1A processing design) and elude the software tests. Remember that code is written by humans, and that the history of code in the satellite era is not encouraging. If such an error exists, and the Level-0 data are not stored, the results will be catastrophic -- **MODIS data can be forever lost.** The risk involved in this decision is so important as to outweigh any user benefits, the number of which at Level-1A is likely to be small anyway. We will make tested unpacking algorithms available on the DADS to any users who desire Level-1A data. Finally, the reduction in storage requirements by leaving Level-1A data packed will result in a significant cost savings over the lifetime of the sensor.

MODIS LEVEL-1A PROCESSING SYSTEM FUNCTIONAL REQUIREMENTS OVERVIEW

Requirements for the functions are derived from the ECS Requirements Specifications document (reference may be found at the end of the data dictionary), and are stated verbatim from the reference.

A. INPUT

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL RECEIVE:

1. Level-0 Data

   (Page 7-23, 3PGS-00440: The PGS shall accept from the DADS L0-L4 Data Sets.)

2. Ancillary Data

   (Page 7-23, 3PGS-00450: The PGS shall accept from the DADS Ancillary Data Sets.)

   (a) Instrument Status Information required for data quality information
   (b) Spacecraft Ancillary Data required for navigation and data quality information

3. Quick-Look Data

   (Page 7-24, 3PGS-00530: The PGS shall generate quick-look products in support of field experiments, event monitoring, and instrument monitoring using algorithms and calibration coefficients provided by the scientists.)

B. CONTROL

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL RESPOND TO CONTROL:

(Page 7-21, 3PGS-00270: The PGS shall provide a scheduler with the capacity to perform the following functions, at a minimum: (a) Add tasks to the job queue, (b) Allocate tasks among processors, (c) Initiate execution of tasks in the job queue, (d) Suspend execution of tasks, (e) Resume execution of a suspended task, (f) Cancel execution of tasks, and (g) Request and verify the staging and/or destaging of data stored in the DADS.)

C. PROCESSING STATUS INFORMATION

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL GENERATE FAULT INDICATIONS:

(Page 7-22, 3PGS-00320: The PGS shall display detected faults to the system operators.)

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL PROVIDE STATUS INFORMATION

(Page 7-22, 3PGS-00380: The PGS shall monitor its internal operations and generate a status report periodically.)

D. OUTPUT

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL PRODUCE:

1. Level-1A Data Products

   (Page 7-13, 3DAAC00070: The DAAC shall generate Levels 1, 2, 3, and 4 data products, archive, manage, quality check and account for archived data products.)

2. Processing Log

   (Page 7-22, 3PGS-00360: The PGS shall generate a PGS Processing Log periodically that accounts for all data processing activities.)

3. Metadata

   (Page 7-24, 3PGS00510: The PGS shall have the capability to generate metadata according to the algorithms provided by the scientists and associate this metadata with each standard data product generated.)

   (Page 7-14, 3DAAC00220: The DAAC shall generate browse data and metadata for routing to the requested users, through the coordination of IMS.)

4. Quick-Look Product (Level-1A)

   (Page 7-13, 3DAAC00050: The DAAC shall provide the ICC with quick-look products for further evaluation of instrument operations and data quality.)

   (Page 7-14, 3DAAC00260: The DAAC shall produce quick-look products for priority transfer to the ICCs.)

   Do these requirements infer that Level-1A Quick-Look Products be sent to the ICC?

E. OTHER

MODIS LEVEL-1A PROCESSING SHALL BE ACCOMPLISHED USING TWO DISTINCT SETS OF STAND-ALONE SOFTWARE: ONE SET TO SUPPORT MODIS-N PROCESSING AND ONE SET TO SUPPORT MODIS-T PROCESSING.

(Unreferenced)

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL BE CAPABLE OF REPROCESSING

(Page 7-24, 3PGS-00540: The PGS shall reprocess specified science data using new and/or updated algorithms provided by the scientists.)

(Page 7-24, 3PGS-00550: The PGS shall reprocess science data using the original or updated (provided by the scientists) calibration coefficients.)

THE MODIS LEVEL-1A PROCESSING SYSTEM SHALL BE CAPABLE OF PRODUCING LEVEL-0 DATA FROM LEVEL-1A DATA

(Requirement inferred from definition of Level-1A data)

MODIS LEVEL-1A PROCESSING SYSTEM FUNCTIONAL REQUIREMENTS OVERVIEW: DATA DICTIONARY

In the following, statements enclosed in quotations are quoted verbatim from the ECS Requirements Specifications document (reference may be found at the end of the data dictionary). Statements not enclosed in quotations are attributed to the MODIS Data Study Team.

Ancillary Data: "Any data, other than standard products, that are required as input in the generation of a standard product. This may include ancillary data from the EOS platforms and the attached payloads, as well as non-EOS ancillary data. All ancillary data are received by the PGS from the DADS." (Page 7-17). For Level-1A, ancillary data includes (a) Instrument Status Information and (b) Spacecraft Ancillary Data. For Level-1A, Ancillary data does not include Locally Maintained Databases.

Anomaly Reports: A report identifying a discrepancy between two or more sources of information. (Unreferenced).

Audit Trail: A record that describes the processing history of data and its identification. Contained within the metadata. (Unreferenced).

Completeness: A data quality indicator determining whether a particular data increment is present in finished form or whether there are missing items. (Unreferenced).

Control: "The PGS shall provide a scheduler with the capacity to perform the following functions, at a minimum: (a) Add tasks to the job queue, (b) Allocate tasks among processors, (c) Initiate execution of tasks in the job queue, (d) Suspend execution of tasks, (e) Resume execution of a suspended task, (f) Cancel execution of tasks, and (g) Request and verify the staging and/or destaging of data stored in the DADS.)" (Page 7-21). In addition are (h) Select processing mode and (i) Request processing status information. Two types of cancel operations are provided: (1) non-graceful (no output generated) and (2) graceful (output up to the cancellation point generated).

Data Quality Check: The process by which data quality information is generated. (Unreferenced).

Data Quality Information: Information on data quality, including existence, completeness, and the presence of anomaly reports, at a minimum. (Unreferenced).

Existence: A data quality indicator determining whether a particular increment of data is present or absent. (Unreferenced).

Fault Indication: An unsolicited flag denoting that a hardware or software error has occurred (e.g., a disk drive failed during data transfer or data header identifiers are not correct), or an "alarm event." (Unreferenced).

Instrument Data: "Data specifically associated with the instrument, either because it was generated by the instrument or included in data packets identified with that instrument. These data consist of instrument science and engineering data, and possibly ancillary data. These data may be assembled for transmission by the instrument, or by an on-board processor of the instrument data." (Page A-9). "Data created by an instrument including scientific measurements and any engineering or ancillary data which may be included in the instrument data packets." (Page A-9).

Instrument Status Information: "High level information about the status of an instrument stored in a designated DADS. These are redundant backup copies only. Primary backup copies are maintained at the ICC." (Page 7-34).

Level-0 Data: "Raw instrument data at original resolution, time ordered, with duplicities [sic] removed." (Page A-4) .

Level-1A Data Product: "Level-0 data, which may have been reformatted or transformed reversibly, located to a coordinate system, and packaged with needed ancillary, engineering, and auxiliary data." (Page A-4). Includes instrument data, a header, and data quality information.

Metadata: "Information which is obtained from data sets, and which provides an understanding of the content or utility of the data set. Metadata may be used to select data for a particular scientific investigation." (Page A-11) Metadata will include an audit trail. (Page 7-18). A more detailed description of the Level-1A metadata product is contained in the Appendix.

Processing Log: "Periodically accounts for all data processing activities." (Page 7-22, 3PGS-00360). A record of the time-ordered processing events. An event may be the completion of the processing activity or the generation of an anomaly report.

Processing Mode: There are three types:

a. Standard Product Processing: "The PGS shall have the capability to produce each standard product as specified in that product's Standard Product Specification." (Page 7-23, 3PGS-00470).

b. Reprocessing: "The PGS shall reprocess specified science data using new and/or updated algorithms provided by the scientists." (Page 7-24, 3PGS-00540).

c. Quick-Look Data Processing: "The PGS shall send the DADS quick-look data for routing to the appropriate destination (e.g., ICC, SCF). Quick-look data shall contain the following information at a minimum: (a) Product identication, (b) quick-look data, (c) associated metadata, (d) process facility identification, and (e) current date and time." (Page 7-30, 3PGS-01260).

Processing Performance: A statement of the amount of data processed; will include a record during processing (dynamic status) and a post-event record (static status). (Unreferenced).

Processing Status Information: "Information regarding schedules, hardware and software configuration, exception conditions, or processing performance." (Page 7-18). The Level-1A Processing System is concerned only with fault (exception) conditions and processing performance.

Quick-Look Data (Level-0): "Real-time or priority playback data which receive minimal processing and are forwarded to the user for his review/use. The user may provide additional processing to suit his requirements." (Page A-14). "Data Received during one TDRSS contact period which have been processed to Level-0 (to the extent possible for data from a single contact). This is data that have been identified as requiring priority processing on the order of a few hours. It is routed to the PGS from the DADS." (Page 7-18). At Level-0, these data are not necessarily time-ordered, complete, nor have duplicates removed, but are at original resolution.

Quick-Look Product (Level-1A): "Quick-look data that has been processed by a PGS prior to being sent to an ICC." (Page 7-35). At Level-1A, Quick-Look Products are not necessarily time-ordered, with duplicates removed, but are at original resolution, and are packaged with necessary ancillary and engineering data. The product is reversible to Level-0 Quick-Look Data. It includes instrument data, a header, but may not have data quality information.

Spacecraft Ancillary Data: "Data available on board a spacecraft, derived from spacecraft parameters, or resulting from the on-board substitution of backup spacecraft parameters, but not produced by an instrument, which are needed for the processing or interpretation of instrument data. Spacecraft ancillary data comprises data referred to as "engineering", "core housekeeping" or "subsystem" data and includes parameters such as orbit position and velocity, attitude and its rate of change, time, temperatures, pressures, jet firings, water dumps, internally produced magnetic fields, and other environmental measurements." (Page A-15).

REFERENCE

Functional and Performance Requirements Specification for ECS, Fourth Preliminary, September 14, 1990.